

## Chapter 4: Result

---

### 4.1. Research Questions

As an effort to reach the desired objectives fulfilling the quality constraints, researches has been done for the following questions, provided in Table 2.

Table 2 : Research Questions

Nu m	Question	Strategy	Answer Path
0	How to develop Autolux application into responsive and maintainable website?	Field, Library, Workshop	4.3 - 4.5
1	What is the current state of the Autolux project?	Field	3.1
2	What is Phalcon PHP and what are the benefits?	Library	4.2.2
3	What is the state of the current website and what are the problems?	Field, Library, Workshop	3.3, 4.3
4	How to implement Phalcon PHP as a solution for the website problems?	Workshop	4.4, 4.5
5	What are the desired SEO (Search Engine Optimization) standards in improving the website?	Field, Library	4.3.2
6	What plugins/add-ons can be used to support a high level “user generated content platform” which also fulfilled the SEO standards?	Workshop	4.4.6
7	What extra feature can be added in order to improve the website?	Workshop	4.4.8, 4.4.9, 4.5.3, 4.5.4

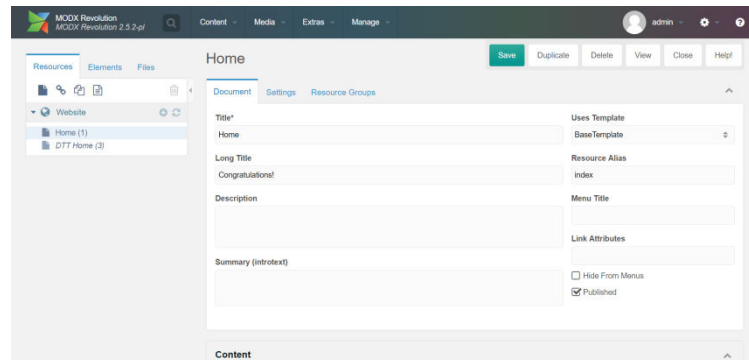
### 4.2. Base Framework

Autolux website was built using *MODx*, a Content Management System (CMS) framework which available as free Open Source. *MODx* still use PHP as its base language while adding some additional new syntax. The company wants to change into PhalconPHP, which is pure PHP framework with Model-View-Controller (MVC) pattern and available as free Open Source. What makes those frameworks different?

#### 4.2.1. Understanding the difference of CMS and MVC

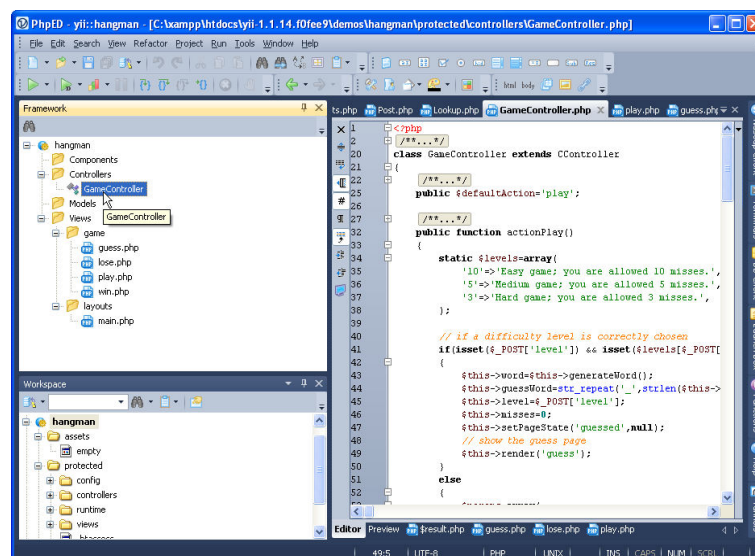
CMS framework is more of an application rather than framework. It provides the user with advance GUI, where it acts as the backend of the website. By using the GUI backend, user may create frontend pages and decorate them. It is easy, fast, and so called all-in-one-package. The downside of using CMS is it has either limited or too-broad features depend on how complex the website is. For example if someone is creating a

static website which contains only 1 page, CMS is far too advance and it will consume more memory. Whereas when the website needs to be dynamic and flexible, like providing statistic for every second, CMS might not be able to follow up.



**Figure 4 : Backend GUI of MODx (CMS framework)**

MVC is structure or pattern commonly used in web development. Model acts as the one who retrieve the data, so it always interacts with the database. View contains mostly of HTML code, and it acts as the interface showed in the web browser. Lastly, the Controller is, as its name suggests, the one which control everything. It connects the Model into View. Beside the structure, MVC frameworks, in general, contain collections of classes and functions (in code), where users can freely use them to build their desired features almost from scratch. By using MVC framework, user has the basic structure of website which is more neat and maintainable rather than just a plain PHP. The downside of using MVC framework is it does not have any Graphical User Interface, so the users have to hardcode everything. Also in practice, the knowledge about Object-Oriented Programming is required. Working with MVC frameworks require source code editor such as PHP Storm, or Notepad++ (there are other tools available).



**Figure 5 : MVC Framework in PHPEd tool (Nusphere, n.d.)**

Even though it is called as MVC framework, it does not mean that MVC is an application or methodology. MVC is just one of many design patterns available. Later on,

MVC was popularize by ASP.NET and become well-known. Many teams started to build their own framework using this pattern, such as Laravel, CodeIgniter, CakePHP, Symfony, Zend, PhalconPHP, Yii, and many more. MVC pattern is considered as a good support for Agile Methodology because there is separation of concerns. Each concern can be developed by different people, making the pattern reusable and extendable. For example, the UI designer and programmer will only work on the view, and they can make multiple types of interface. As for the database and system programmer will focus on building the Model. They will meet at the Controller. This pattern also supports the element of maintainable because it has a tidy and structural separation. The next programmer who handles the website will definitely know that HTML code can only be found in the View, whereas logic or database retrieval can only be found in Model.

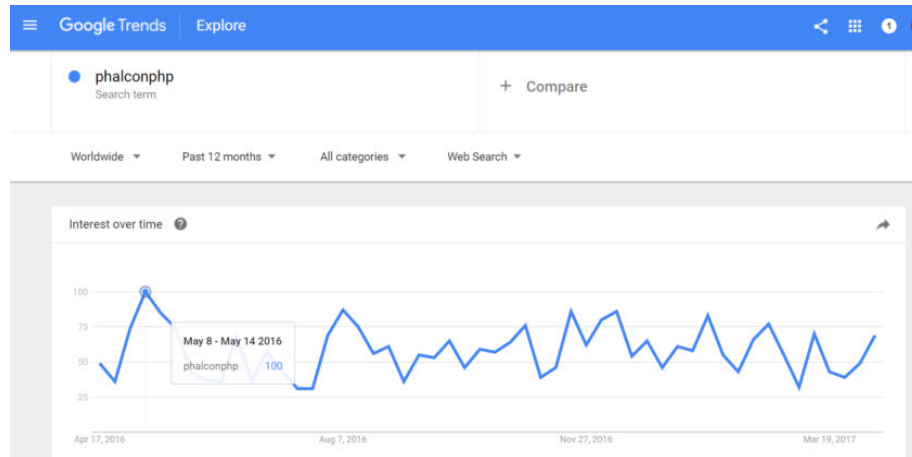
In this project, the company prefers to use MVC because they are planning to build as many plugins as possible to support their creations independently (not depending on open-source plugins). DTT Multimedia thought that if they use a ready-made application, the only benefit is fast delivery. And the website product will have many limitation which leads to the element of inextensible. Another downside is that the CMS system itself is consuming memory far greater than using MVC framework.

#### 4.2.2. Why PhalconPHP?

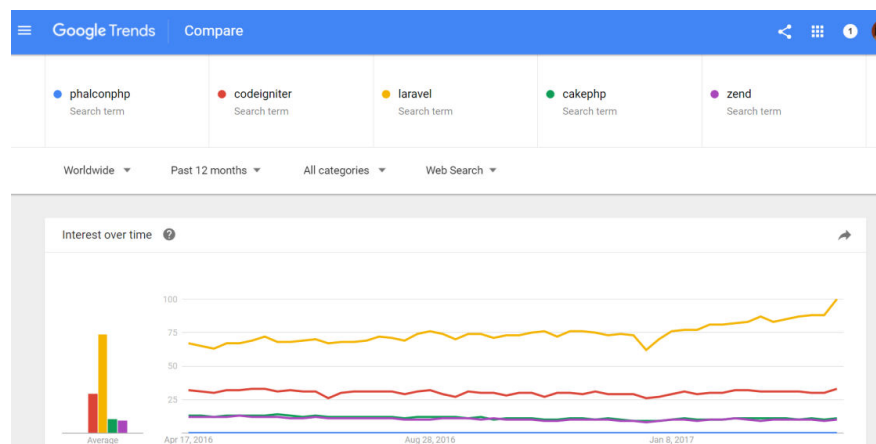
PhalconPHP is just one of many PHP MVC frameworks available. Based on the 1 year search on Google Trends taken in April 14, 2017 (12:29 +0200 UTC), the highest number is 100 search in a week (see Figure 6). Compared to CodeIgniter, Laravel, CakePHP, and Zend, the popularity of PhalconPHP is very low (see Figure 7). For more detail about popularity comparison of these 5 frameworks, please look into Appendix 2: PHP Frameworks Comparison.

It cannot be denied that Laravel and CodeIgniter are leading in PHP Frameworks. Laravel can be easily generated from Composer, and CodeIgniter only needs to be copy-pasted. Both frameworks will work just fine on the server since they run on Zend engine (the default engine on server). On the other hand, PhalconPHP run on its own engine which needs to be installed first on the server.

Both Laravel and CodeIgniter are rich in advance libraries, while PhalconPHP focus on creating basic library. For example the mail library is not available in PhalconPHP. Another example is Laravel also has its own template engine, named Blade, which help in building a cleaner HTML. This feature is also available in PhalconPHP with the name Volt. Another strength of Laravel is unit test. Until this report was written, Laravel is the only open source PHP MVC framework which has this feature. It can be said that Laravel is indeed the best framework available now, but the project was built on PhalconPHP because the company insist on using PhalconPHP. DTT Multimedia has been using PhalconPHP for more than 2 years now.



**Figure 6 : PhalconPHP popularity on Google Trends**



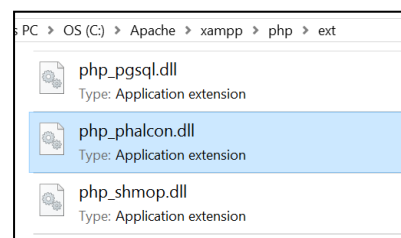
**Figure 7 : Frameworks' Popularity Comparison (Phalcon is the blue line)**

Phalcon was released on November 14, 2012. The official website is located at <https://phalconphp.com/en/>. PhalconPHP is delivered as a C extension. What is C extension? Mentioned by Jonathan Leffler (2011) that:

*“An extension in C is a feature supported by a C compiler that is not described (permitted, mandated) by the C standard (and similarly for C++, since the question is tagged like that). For example, GCC allows you to define a function within the body of another function; standard C does not. That is a C extension.”*



**Figure 8 : PhalconPHP logo (Marco Platanic, 2017)**



**Figure 9 : Phalcon as C extension, located in php directory**

Compared to PHP, C is considered as a low level language. Building an application using C will take much longer time, but the bright side is that C has the ability to manage memory and the compiling speed is much faster because C is closer to machine language (compared to PHP). So, with the idea to gain faster compiling time, Phalcon was built using C and able to interpret PHP. How can PHP code tie into C? In fact, PHP is written in C. Despite being a C extension, PhalconPHP has the ability to run under Zend Engine, *“the open source scripting engine that interprets the PHP programming language.”* (Wikipedia 2016, par. 1)

But of course PhalconPHP has the upper hand for speed, since it is adopting the C language. As mentioned by Al Rooker (2016, par. 2):

*“PHP is an interpreted language - the interpreter has to read your code and make sense of it before it executes, which adds an infinitesimal amount of time to the process. Phalcon is a framework compiled in C. So when your own code hooks into a class/method of Phalcon, that method will be executed directly as C, which is already compiled to machine code, which is basically as fast as you can get.”*

Thanks to its low-level architecture and optimizations, PhalconPHP provides the lowest overhead for MVC-based applications. Low overhead means that the framework is low in memory consumption and CPU compared to other frameworks. So, is PhalconPHP winning the competition of PHP frameworks? The answer is not yet, because:

1. The stable release (version 3.0.3) was published on December 24, 2016; which is just recently (this is according to Wikipedia 2017). This means that their popularity is still starting up.
2. Unlike most PHP frameworks where user only need to copy-paste the files to server, Phalcon needs to be installed first in the server before using it. The reason behind it is because Zend Engine cannot understand the Phalcon's C language.
3. Phalcon is the join of C and PHP. For PHP programmers who never experience C language, Phalcon will not be a good choice as a PHP framework starter.

But there will be a big chance where PhalconPHP will soar high. Being the fastest framework, PhalconPHP will be a very good option for websites with heavy visitor traffic or heavy application. Here are some more advantages of PhalconPHP:

1. To support database, Phalcon is using PHQL. Mentioned in the Phalcon 3.1.1 documentation (n.d., par. 1) that:

*“Phalcon Query Language, PhalconQL or simply PHQL is a high-level, object-oriented SQL dialect that allows to write queries using a standardized SQL-like language. PHQL is implemented as a parser (written in C) that translates syntax in that of the target RDBMS (Relational Database Management System).”*

By using PHQL, executing query will be faster. It also allows user to define the relations between tables, by using `hasMany` and `belongsTo` syntaxes. It has query builder for shorthand in creating query command. For example, joining table:

```
$this->modelsManager->createBuilder()->from('Tbl1')->join('Tbl2');
```

For complex queries which require subquery or using uncommon command (like DATEADD, INTERVAL, HAVING, etc.), it has to be typed as raw SQL query.

2. Volt is the template engine which is highly integrated with PhalconPHP. The syntaxes are very clean and have many similarities with Twig, another template engine which is more popular. Volt is automatically provided when installing PhalconPHP, and very recommended to use since it is lightning fast, highly functional and well documented. Written in Sitecrafting (2014):

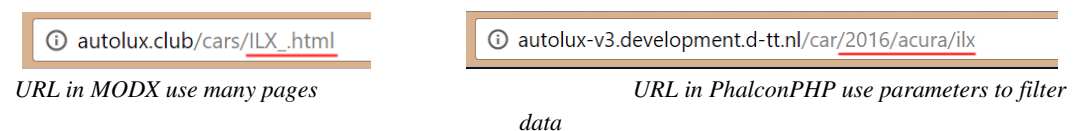
*“With this engine running in compiled-C, Volt’s non-cached speed soars into supersonic territory (23, 906 templates/sec) and doubles this speed when leveraging cache (47,851 templates/sec). Most impressively, it pulls off those numbers with the entire Phalcon Framework stack load included. As you may recall, the other engine bolted to a framework (Blade) was roughly 20x slower than this after including its whole framework.”*

3. Syntax structure in PhalconPHP is closely similar to C language. The usage of namespace and use commands makes the user able to locate the source code.
4. Phalcon Forum is available for any questions regarding the framework.

## 4.3. Autolux Website

### 4.3.1. Migrating from MODx to PhalconPHP

The big difference of using MODx compared to PhalconPHP is that in MODx, user is creating pages as HTML files (.html). Whereas PhalconPHP only create pages as template, and retrieve the data from database to fill out the content. The easiest example is the car detail page (see Appendix 6: Screenshots of Pages).



**Figure 10 : URL Comparison**

In MODx, there are ILX\_.html, DS6WR.html, and many more. Each page is representing a car, so if there are 120 cars, then there will be 120 HTML files. By using PhalconPHP, there will only be 1 PHP page used as the base template of car detail. As for the content, it will be retrieved from database depend on the parameters in URL. In this way, the home page can loaded faster because of there are less files and when there is new car, the page will be auto generated after inserting. So, will migrating from MODx to PhalconPHP be possible? The answer is yes, but it will take time since the only parts which can be used from MODx are the interface parts (HTML, CSS, and JS).

### 4.3.2. Search Engine Optimization Implementation

Search Engine Optimization (SEO) is the life of DTT websites now. Why is it so important? According to Rand Fishkin in Moz website (2015, par. 1):

*“SEO is a marketing discipline focused on growing visibility in organic (non-paid) search engine results. SEO encompasses both the technical and creative elements required to improve rankings, drive traffic, and increase awareness in search engines. There are many aspects to SEO, from the words on your page to the way other sites link to you on the web. Sometimes SEO is simply a matter of making sure your site is structured in a way that search engines understand. SEO isn't just about building search engine-friendly websites. It's about making your site better for people too.”*

Google will also give subheadings (also known as site links) to the good website, making users able to jump on certain page or section. But these site links are automatically generated by Google if the website is really popular. Even if it is automatically generated, it does not mean impossible, the website just have to gain more traffics and gaining more popularity. And that is the SEO will do the work to boost the website's popularity. By implementing SEO, the website will gain higher search traffic and be more noticeable to internet users.

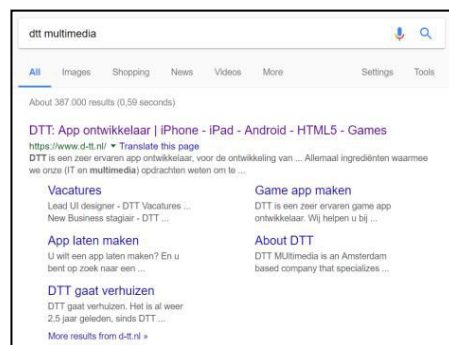


Figure 11 : Google Subheadings/Site Links

DTT Multimedia knows the best about how crucial the value of SEO for their product, which is why they have implemented some standard. These are the standard values given by DTT on implementing good SEO:

- All should operate responsive (mobile/tablet/desktop).
- All should be delivered by serious SEO standards, which are:
  - SEO friendly URL (no capitals)
  - Clear URL structure
  - Non-www version
  - URL without any prefix
- Option to insert meta description for all pages automatically and make it editable
- All pages should have a page title automatically: Autolux | Ferrari F40 (H1: Car brand + Name)

- H1, H2, H3, tags etc.
- Connect with Google Analytics and set events to relevant conversion buttons
- Page speed must be high, while the loading time must be low
- Registered in Google Webmaster Tools
- Create sitemap.xml and robots.txt
- Images should be optimally resized for fast loading, receive good titles, and automatic alt text.
- Add some text on homepage. The text should contain h1, <strong>, <em>
- Set up keywords map in footer

## 4.4.Frontend Building Process

### 4.4.1.Preparation

The old Autolux has been running at <http://autolux.club/> for 2.5 years. The first thing to do is downloading, and studying the old system. The download phase has no problem, except that it had a very huge memory size because of the high resolution images. As for the studying phase, it took 2 days longer because the database structure of MODx was confusing and unstructured. There is no clear documentation in the whole project, and multiple consultations with other developers were held.

### 4.4.2.Building Header and Footer

In the new website, the layout will be separated into 3 parts: head, body, and foot. Head section is exclusively for <head>, while the <body> will be separated into body and foot. The content of body will be retrieved based on the action triggered from controller. While the content of the body will always change depends on which page is access, the content of head and footer will never change.

```

1 <!DOCTYPE html>
2 <html>
3   <!-- The head -->
4   <head>
5     <meta charset="utf-8" />
6     <title>Autolux | Home</title>
7     <meta http-equiv="X-UA-Compatible" content="IE=edge">
8     <meta content="width=device-width, initial-scale=1" name="viewport" />
9     <meta name="description" content="The largest car platform which collects the latest models and the best vintage cars" />
10    <meta name="author" content="DTT Multimedia Amsterdam" />
11    <meta name="keywords" content="car platform, dtt multimedia, amsterdam">
12
13    •
14
15    <!-- CSS -->
16    <link rel="stylesheet" type="text/css" href="/css/frontend/form.css" />
17    <link rel="stylesheet" type="text/css" href="/css/frontend/kootam.css" />
18    <link rel="stylesheet" type="text/css" href="/vendors/frontend/DTTcomment/dtt.comment.css" />
19
20    <!-- jQuery (necessary for Bootstrap's JavaScript plugins) -->
21    <script async src="https://ajax.googleapis.com/ajax/libs/jquery/1.11.3/jquery.min.js"></script>
22
23  </head>

```

Figure 12 : Head part



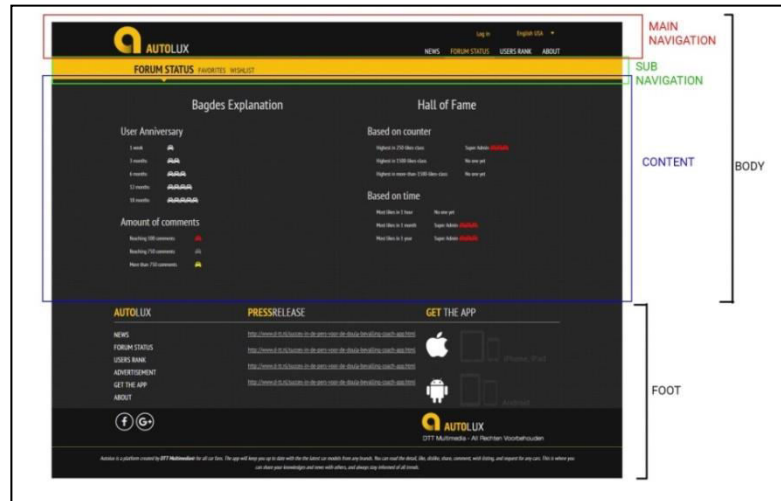


Figure 13 : Body and Foot Parts

To support the SEO, footer sitemap and keyword mapping is provided in the lowest part of the site. The footer sitemap is not the sitemap.xml; it is just a list of links which will redirect user to certain page. Sitemap.xml is file located in root directory. Both sitemap footer and sitemap.xml serve to help visitors and search engine spiders to find information regarding what pages are provided.

Several words located in the footer is called keyword mapping. In the old time, developers usually add a line with a bunch of keywords (with random words too) to make the website more “noticeable” in the search engine. But putting so many random words in order to manipulate the site’s ranking, lead to a new rule established by most search engines: banning the website with “keyword stuffing”. What will be referred as “keyword stuffing”? Based on Google Webmasters Support (n.d.):

- Word which is repeated too often (to the level of unnatural phrasing) in a page will be considered as spam, and the search engine will ban the page.
- Lists of phone numbers without substantial added value.
- Blocks of text listing cities and states a webpage is trying to rank for.

Words which are hidden with CSS “display:none” or font color equals background color, will also be regarded as spamming attempt. And putting keywords in the footer decrease the design value. A solution given by many developers from the internet is to focus on putting the keywords on the content, which looks natural. But then it goes against the company’s demand. The solution reach from reviewing 20 website footers is to make 3-4 sentences which contains the keywords.



Figure 14 : Autolux Website Footer

### 4.4.3. Building Index, Favorite, and Wishlist Pages

These three pages have similar HTML interfaces; the difference relies on which data is retrieved. SQL queries for retrieval are written in the models. PhalconPHP supports the feature to execute query from controller, but because the queries are using subquery and uncommon command, it has to be written as raw query in the model.

For the views, those pages are using `{{partial()}}` from Volt. “Partials” is the same as “include”. By using this method, there is a separation of concerns (so the content is separated into `container.php` and `thumbnail.php`). `Thumbnail.php` will act as a template. Inside the container, there are repetitions of thumbnails, see Figure 15. Another advantage is that if the thumbnail template needs to be changed, the developer will only change related file.

```
<div class="container">
  {{ partial("thumbnail", ['car': car1]) }}
  {{ partial("thumbnail", ['car': car2]) }}
  {{ partial("thumbnail", ['car': car3]) }}
  {{ partial("thumbnail", ['car': car4]) }}
</div>
```

**Figure 15 : Partialing Thumbnail**

Two of the SEO standards given by the company are “Page speed must be high” and “Page loading time must be low”. In order to reach these standards, the method chosen is the infinite scroll. A plugin named `jScroll` is used, mentioned by Philip Klauzinski (2017, par. 1):

*“jScroll is a jQuery plugin for infinite scrolling, written by Philip Klauzinski. Infinite scrolling; also known as lazy loading, endless scrolling, autopager, endless pages, etc.; is the ability to load content via AJAX within the current page or content area as you scroll down. The new content can be loaded automatically each time you scroll to the end of the existing content, or it can be triggered to load by clicking a navigation link at the end of the existing content.”*

By using `jScroll`, the page will only need to load the first page, followed by the link in the next page. When the link is entering the viewport, the next content will be loaded. It will continue until the next page return empty. This method will decrease the first load time.

### 4.4.4. Building Filter Feature

Cars, which displayed in home, favorite, and wishlist pages, can be filtered by brand. There are 3 options proposed by the developer to the company:

1. Providing a filter page with brand parameter. With jQuery, the filter page will overwrite the content of current page. The downside of this method is that there is a chance that the load function can take several seconds to finish the loading because the page is really heavy.

2. Eliminating unused cars with jQuery by setting the CSS to display:none. The downside of this method is that it only eliminates the cars which has displayed.
3. This method is combination of the previous methods. After eliminating unused cars, the next loaded cars will be filtered first before displaying. The downside of this method is there will be many jQuery codes running together which can cause crash.

Based on the available, the company chose the first option because the chance for slow load is very thin since 1 time load will only displayed 10 cars.

#### 4.4.5. Building Car Detail Pages

Car detail pages are actually only a page, we named it detail.phtml. By using detail.phtml as a template, the data will be filled out in some part, resulting in a different page. The indicator of which data to retrieved is the URL. Using the GET method, it will catch the parameters, and search the matching data. The URL is design to be friendly to make it human readable, so sharing the URL would be easier. This was done to support SEO, too. And there are also restriction and checking, to prevent any SQL injection. If there is no matching data found, it will automatically show the error 404 page.

If there is any ID which needs to be printed, the ID will be encoded first. In the car detail page, user may like/dislike the displayed car, put it on wish list, or share in on their social media. This system is created to simplify user in sharing their favorite car and gain more popularity of the website.

#### 4.4.6. Building Comment Plugin

In the Autolux website, there will be comment section on each car article. For easy integration and SEO support, there are several service available; such as Disqus ( <https://disqus.com/> ). Disqus offers free add-on tools for website to power discussions, increase engagement, and earn revenue. The implementation only needs registration into Disqus, and copy-paste a few code into the website. Disqus also provide the analytic page, which can be accessed after registering. Disqus premium version supports SEO by making each comment searchable in the search engine and shareable in the social media. The downsides of using free version Disqus are it does not support self-single-sign-on (website's login), does not support anonymous user, and sponsored advertisement cannot be removed. Due to the demand of the company for being independent, research was conducted.

The company never demands for a plugin. But it would be nice if it was implemented as plugin rather than just a feature. The client gave the permission to extend the deadline because the idea will benefit them. It was decided to create the comment plugin as jQuery plugin, with the reasons:

1. jQuery has AJAX which can communicate with the server side programming.

2. jQuery has many methods which support animation to make the interface more interesting.
3. jQuery plugin can be easily integrate in HTML. With only few lines of code, it will set up both server-side (PHP code) and client-side (HTML code) itself.
4. jQuery plugin can be set to run after the page is loaded, therefore important information can be delivered first.

#### 4.4.6.1.Data Structure

The research began from how to store data efficiently as a nested hierarchical database. There are 4 methods given as solutions by Bill Karwin (2010) on his presentation, but only 3 match the requirements:

- Adjacency List

Adjacency List is similar with Singly Linked List. But Singly Linked List only permits 1 node, while Adjacency List is open to as many as possible. It is the simplest method available to create chain data, but not very good method for hierarchical data. The child will only know its immediate parent (the parent right above the child), so its depth is unknown. When retrieving the whole hierarchy in loop, the end condition is also unknown. Please look at the Figure 16.

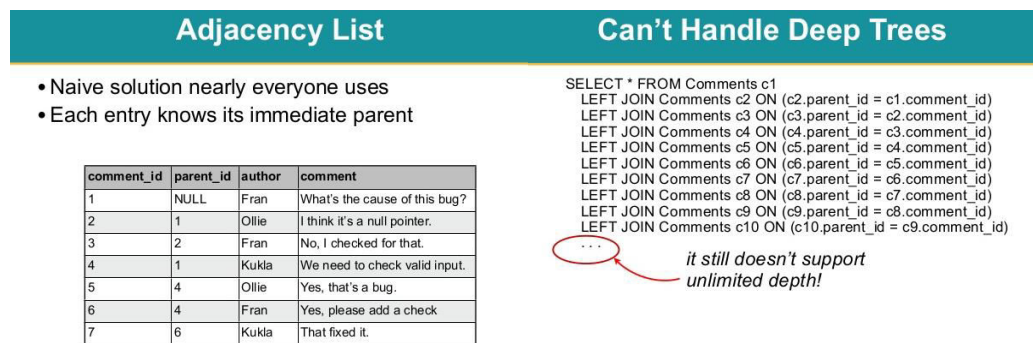


Figure 16 : Adjacency List (Bill Karwin, 2010)

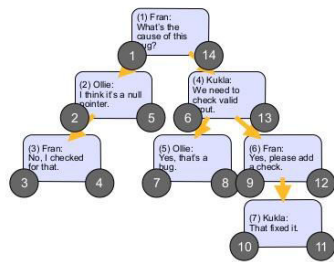
- Nested Sets

Nested set is the most complex out of 3 methods. Explained by Wikipedia (2017, par. 4):

*“The nested set model is to number the nodes according to a tree traversal, which visits each node twice, assigning numbers in the order of visiting, and at both visits. This leaves two numbers for each node, which are stored as two attributes. Querying becomes inexpensive: hierarchy membership can be tested by comparing these numbers. Updating requires renumbering and is therefore expensive.”*

This is a very good solution to store comment database, but the problem lies in the insert query. Inserting new comment means that the value stored in comments' left and right need to be re-evaluated from top to bottom.

## What Does This Look Like?



## What Does This Look Like?

comment_id	nsleft	nsright	author	comment
1	1	14	Fran	What's the cause of this bug?
2	2	5	Ollie	I think it's a null pointer.
3	3	4	Fran	No, I checked for that.
4	6	13	Kukla	We need to check valid input.
5	7	8	Ollie	Yes, that's a bug.
6	9	12	Fran	Yes, please add a check
7	10	11	Kukla	That fixed it.

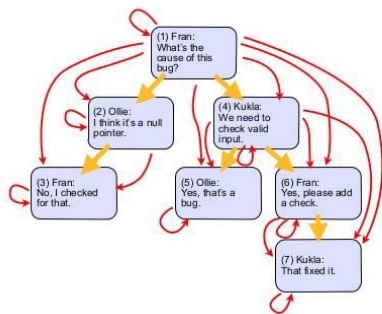
Figure 17 : Nested Sets (Bill Karwin, 2010)

- Closure Table

Closure table utilizes the many-to-many relationship in database. It provides another table which records the relationship of parent and child. A parent can have multiple records since it has many children. This method consumes more memory than others, but the best one so far.

## Closure Table illustration

## What Does This Look Like?



comment_id	author	comment
1	Fran	What's the cause of this bug?
2	Ollie	I think it's a null pointer.
3	Fran	No, I checked for that.
4	Kukla	We need to check valid input.
5	Ollie	Yes, that's a bug.
6	Fran	Yes, please add a check
7	Kukla	That fixed it.

requires  $O(n^2)$  rows  
(but far fewer in practice)

ancestor	descendant
1	1
1	2
1	3
1	4
1	5
1	6
1	7
2	2
2	3
2	4
2	5
2	6
2	7
3	3
3	4
3	5
3	6
3	7
4	4
4	5
4	6
4	7
5	5
5	6
5	7
6	6
6	7
7	7

Figure 18 : Closure Table : (Bill Karwin, 2010)

Between those 3 methods, Adjacency List was picked. A recursion is added in PHP code to retrieve each depth. By using recursion, the depth can be unlimited, but it is not recommended to use for more than 4 level since the hierarchy will be confusing to readers and there is a chance that the SQL failed to retrieve in certain depth, resulting in crashed data.

### 4.4.6.2.Security & Anti-Spam

In implementing the plugin, some security standards are given, such as hashing each comment ID and spam prevention. For the spam prevention, all inputs were given special captcha. The ordinary captcha, where the user needs to answer the question given, is considered old and design-unfriendly. For highly security reason, some websites keep putting the ordinary captcha in. But since this website will value design more than tight-security, “honeypot captcha” was implemented. Comment spam bots usually will fill

every input field which is available in the form and usually to gain more speed, they will skip the CSS files. Explained by Phil Haack (2007, par. 4):

*“To exploit this, you can create a honeypot form field that should be left blank and then use CSS to hide it from human users, but not bots. When the form is submitted, you check to make sure the value of that form field is blank. For example, I’ll use the form field named body as the honeypot. Assume that the actual body is in another form field named the-real-body or something like that:*

```
<div id="honeypotsome-div">  
If you see this, leave this form field blank and invest in CSS support.  
<input type="text" name="body" value="" />  
</div>
```

*Now in your code, you can just check to make sure that the honeypot field is blank”*

Even though it is a type of captcha, it works in the contrary: when a person is sending the form, then he/she does not need to answer/input anything; meanwhile the bots will input something.

#### 4.4.6.3.Special Features

The features are like-dislike comment, report inappropriate comment, editing, attachment, and anonymous user. These features are proposed by DTT Multimedia, as the client, except for the anonymous user. Comment attachment is built to support “user generated content” element, and is available to bind one of these three media: link, image, or video. All media can be referred by using URL, and later on displayed in embedded box. Image can also be attached by uploading from user’s device. Anonymous user is an optional feature which can be turned off in case the website owner wants less possibility of spams. Anonymous comment can be important to enrich the content keywords, which is the very reason to boost SEO.

#### 4.4.7.Building Login and User Access

There are 3 access levels in this website: visitor, member, and administrator. Visitor level is for those who did not register themselves to the website. Visitor can see the cars and share it on social media. Member level is for those who register themselves to the website. They have the optional registration for notification, favorite, wish list, and like/dislike car. After registration, member will receive a validation mail. This is a standard feature to maintain data quality, based on OWASP. Only after the validation can the user login. There is a shortcut for validation, and that is using Facebook Login. Facebook Login will enable user to immediately register and use the website’s functionalities. Administrator level is for those who manage the website, the only level which can access to backend.

#### 4.4.8. Building Multilingual System

There are many methods to create multilingual website. The easiest one is using translation bots, for example Google Translate Add-ons, to replace the website content with its robotic translation. This is not considered as a good practice, for the translation will rough.



Figure 19 : Language Options

Another solution is by providing languages directories in Views, and then developer should create a system to routing user into the chosen language directory. This is “subdirectory” method. The routing indicator used is by retrieving (GET method) the language value written in the URL parameter. After retrieving the language, the controller will pick the view from that language directory.

Alternate solution is called “subdomain” method (for example <http://en.domainname.com/>). Each subdomain will redirect into its own directory of website files, with different translation. This the most advance and tidy method, but not all hosting service have their subdomain feature activated. This method is chosen by the company because the server can have subdomains and additional database system is added. So, all subdomains will be pointing on the same website files directory, but all the texts will be stored inside database. The view will retrieve the texts based on the current subdomain’s name. See Figure 20 for more details.

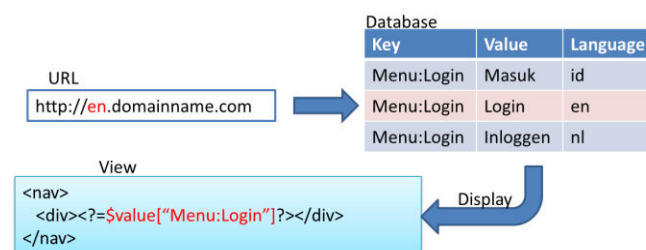


Figure 20 : New Subdomain System

#### 4.4.9. Building Website Push Notification

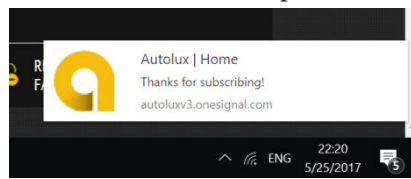
Push notification has been used these days to replace email notification. Compared to email which stays longer, the user feels the urgency to read the information on push notification. Sometimes emails are also trashing the inbox, making user mark the



sender address as spammer which result in rejection of information, which is not good for the website owner.

Autolux will be using web push notification service provided by other server. The reason for using a service is because it is impossible to build your own push notification service without decent resources and planning. It will take a big team to build and an independent server as the workspace. Because of the lack of resources, it was decided to use one of the free-service available in the internet. The chosen provider is OneSignal, addressed at <https://onesignal.com/> , because of several reasons:

- It supports both HTTPS and HTTP, while most service will only support HTTPS.
- Clear documentation available at <https://documentation.onesignal.com/v3.0/docs> .
- It supports action button which can be the potential of further development.



**Figure 21 : Autolux Notification Box**

The notification will be pushed when there is a new comment or a new car posted. To make it maintainable, a PHP class is defined with functions to interact with OneSignal. This class was implemented as a shared plugin in PhalconPHP so that any controller can gain access. The decision of making it as plugin was decided by the developer since it will be easier to reuse and maintain it.

## 4.5.Backend Building Process

### 4.5.1.Choosing the Interface

DTT Multimedia does not have its official interface for the backend, which is why a research is conducted to get the best designed interface; by comparing reviews from 13 websites (full voting can be seen in Appendix 4: Voting Result of the Best Admin Interface). Metis Admin and SB-Admin-2 gains the most votes. Because both has the same number of vote, it was decided to use SB-Admin-2 because it was released way earlier than Metis Admin, which indicate that it is more stable because it has gone through many bug reports.

### 4.5.2.Building Database Management

In the old website, the developer was unable to map the structure of the database, when finally it comes to realization that all the contents are saved into 1 table named “modx\_site\_content”. There is no structure on it; everything was just dumped into that table. Therefore MODx database cannot be used and another study was conducted on the



whole Autolux project (including the mobile application). In the end, it was decided that the current project will be using the mobile application database, and it will be extended to fulfill the demands.

Database manager is the feature to do CRUD operation toward the recorded database. It is simpler than PHPMysqlAdmin, because it is designed for beginners. The “Read” operation is presented using jQuery plugin named DataTable. DataTable is very easy to implement, responsive, and interactive. It also has searching, sorting, and pagination features built in. The “Create” and “Update” operations will redirect user into a form with inputs. As for the “Delete” operation is provided as a trash button on each row in the table.

Advertisement manager is a new idea proposed by the developer. It has the same functions and interface as the database manager for the backend. Each time the frontend car detail page is loaded, the system will randomly select an advertisement stored in database and displays it with the hyperlink automatically.

### 4.5.3. Building Social Media Management

Social media manager is a new idea proposed in order to help company in managing its social media. A list of input was given with the symbol of social media on the right side. The URL of social media will be stored in database, and will automatically be generated on the frontend footer.

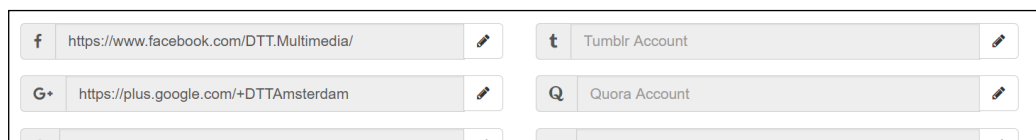


Figure 22 : Social Media Inputs

### 4.5.4. Building Sitemap Management

Sitemap is very crucial in SEO since it helps search engine to crawl the website. It functions as a map, to mark each pages inside the website. Because Autolux is a website with many cars on it, the sitemap has to be created by the developer herself, not using the free limited-service available in the internet. The sitemaps was divided into crawl-daily (for car details page) and crawl monthly (for the rest pages), and were generated using PHP (with the output of XML files). The crawl-daily requires connection to database to retrieve all cars recorded which the very reason of why choosing PHP as the generator language. Sitemap has only one kind of structure, which is open information throughout the internet (for more information, visit <https://support.google.com/webmasters/answer/183668?hl=en> to get in depth). The structure only consists of a header, while the rest are repetitive <url> with different values (see Figure 23 for example).

To edit or re-generate the sitemaps instantly, there will be a page in the backend which has the text editor to easily edit the XML file, and a button to execute the PHP file to re-generate the XML. This is a new idea proposed and accepted by the company.



```

<?xml version="1.0" encoding="UTF-8" ?>
<urlset xmlns="http://www.sitemaps.org/schemas/sitemap/0.9"
  xmlns:image="http://www.google.com/schemas/sitemap-image/1.1"
  xmlns:xhtml="http://www.w3.org/1999/xhtml">
  <url>
    <loc>http://localhost/autolux_v3/car/2016/audi/rs5-cabriolet</loc>
    <lastmod>2017-04-03T14:33:12+02:00</lastmod>
    <changefreq>daily</changefreq>
    <priority>0.7</priority>
    <xhtml:link rel="alternate" hreflang="en" href="http://localhost/autolux_v3/car/2016/audi/rs5-cabriolet/en"/>
    <image:image>
      <image:loc>http://localhost/autolux_v3/img/frontend/assets/images/cars/Audi/RS5%20Cabriolet/image1.jpg</image:loc>
      <image:caption>Autolux Cars - Audi RS5 Cabriolet</image:caption>
    </image:image>
  </url>
  <url>
    <loc>http://localhost/autolux_v3/car/2016/audi/rs5-coupe</loc>
    <lastmod>2017-04-03T14:33:12+02:00</lastmod>
    <changefreq>daily</changefreq>
    <priority>0.7</priority>
    <xhtml:link rel="alternate" hreflang="en" href="http://localhost/autolux_v3/car/2016/audi/rs5-coupe/en"/>
    <image:image>
      <image:loc>http://localhost/autolux_v3/img/frontend/assets/images/cars/Audi/RS5%20Coupe/image1.jpg</image:loc>
      <image:caption>Autolux Cars - Audi RS5 Coupe</image:caption>
    </image:image>
  </url>
  <url>
    <loc>http://localhost/autolux_v3/car/2016/audi/rs6</loc>
  </url>
</urlset>

```

Figure 23 : Sitemap with Daily Crawls

## 4.6. Testing the Website

### 4.6.1. Speed Testing

Based on the SEO standard given by the company, the website must be load in the fastest way possible. What is considered as fast speed? Based on John Steven (2016, par. 9):

*“The golden question is: how fast should my website load in 2016? In Google’s Site Performance for Webmasters video, Maile Ohye, states that “2 seconds is the threshold for e-commerce website acceptability. At Google, we aim for under a half second.” A half second (0.5 seconds) is equal to 500 milliseconds (ms), the literal blink of an eye is 300 to 400 ms. The answer to our question is: from 500 milliseconds to 2 seconds at most.”*

The target speed will be 0.5 to 2 seconds on load. Several testing website were used and the average time to load the homepage is 2.6378 seconds. A little bit over the target due to unoptimized images, but the company’s demand for a good quality image so the website is kept as it is. The testing results in detail can be read in Appendix 5: Speed Test Result.

### 4.6.2. Responsive Testing

Autolux website has been design to be responsive since the old website, with the support of Bootstrap. In this project, refinements are added to minimize the jQuery thus

reducing the time load. The website is put on <http://ami.responsivedesign.is> for testing and has been proved to be responsive.



Figure 24 : Responsive Testing

### 4.6.3. Mobile Application Synchronization

Since the mobile application is available, the website needs to be synchronized. After migrating the API (in this project, API acts as service to provide JSON response), the test was conducted and proved to be synchronized. The only problem encountered is the URL for retrieval in the Android was written in wrong format (for example “ForumStatus” should be called as “forum-status”). The problem was solved by changing the methods in Android code into the right one.

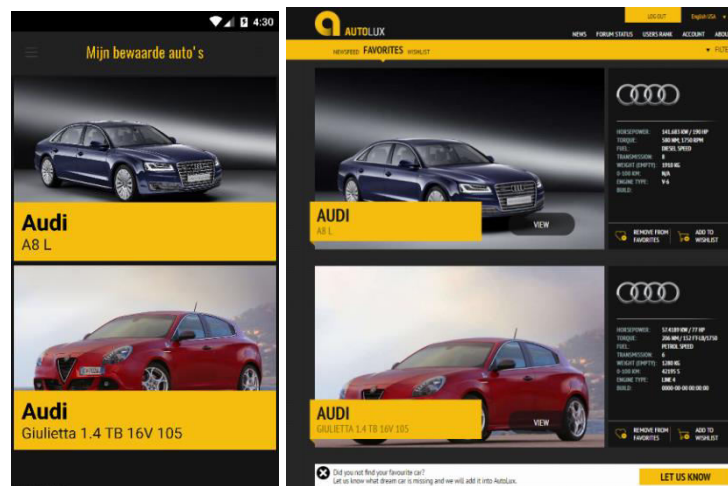


Figure 25 : Autolux Android has the same cars as Autolux Website